Amendments to the Specification:

Please replace the paragraph at page 10, lines 6-14 with the following amended paragraph:

In a preferred embodiment, the software tool is implemented as a JMX Mbean. Java Management Extensions (JMX) is a Java standard that defines a management architecture, APIs, and management services. JMX also defines manageable resources and associated management beans, or MBeans, that provide instrumentation of these managed resources in a ~~standardize6d~~ standardized way. Further, since JMX is a standardized mechanism, existing management tools that are JMX compliant are able to remotely manage resources via the MBeans registered with a JMX agent (generally running in the server). In fact, there is an evolving standard for remote JMX access.

Please replace the paragraph at page 14, lines 1-11 with the following amended paragraph:

In the example shown in Fig. 3, interface 300 includes a plurality of data selection tabs, such as tabs ~~302A-D~~ 306A-D. Each tab represents a type of application and/or application server parameters that are to be tuned. Selection of a tab caused interface 300 to display information relating to the type of parameters represented by the tab, such as the current values of the parameters, in a parameter panel 302. Performance measurements may also be displayed in a measurement panel 304. For example, selecting database tab 306A causes display of parameters related to the database, such as database connection pool size 308, in parameter panel 302. In addition, parameter panel 302 includes controls that allow the values of the displayed parameters to be modified, such as change button 310.

Please replace the paragraph at page 18, line 17 to page 19, line 18 with the following amended paragraph:

In the example shown in Fig. 4, memory 408 includes HTTP listener layer 108, virtual path manager 109, application server layer 110, applications layer 112, ORB 114, ~~code generation objects 410~~ performance measurement objects 411, ~~SAX objects~~ interface objects 412, and operating system 414. HTTP listener layer 108 is made up of listeners, the adapter interface, and dispatchers. Listeners are HTTP servers; they handle incoming requests and route them to the dispatcher. The dispatcher forwards requests to the virtual path manager 109. The virtual path manager maps a request to a cartridge type and passes this information back to the dispatcher. The virtual path manager also passes back authentication requirements to the dispatcher. The Application Server layer 110 provides resource management in handling requests for applications deployed as cartridges on the server. It provides a common set of components for managing these applications. These components include load balancing, logging, automatic failure recovery, security, directory, and transaction components. The Applications layer 112 is made up of applications, cartridges, and cartridge servers. Applications and cartridges are the two main objects that you use when building applications for the application server environment. ORB 114 acts as the middleware between clients and servers. Performance measurement objects 410 determine the values of performance measurements related to application tuning. Interface objects 412 implement the interface shown in Fig. 3. Operating system 414 provides overall system functionality.

Please add the following NEW paragraph at page 27, line 10:

Distributed services management level 506 also includes Web browser 522. Additional management API's are also shown.

Please replace the abstract with the amended abstract attached hereto on a separate sheet.